Remarks

Reconsideration is respectfully requested.

Claims 1-23 were previously pending in this application.

Independent claims 1 and 11 are amended.

In brief, the invention is specialized test software operating on an embedded processor that creates one or more test workers or threads, each having a specific routine to perform, which are executed in parallel, stressing various communication paths. The results may be analyzed to help in many different ways during the life cycle of a device containing the embedded processor.

The test software is a standardized test suite that is capable of exercising all of the input and output ports of an embedded processor, regardless if those input and output ports are used in a specific application. The test software may be preconfigured to exercise each of the input and output ports in various manners so that when powered up and operating, an engineer may diagnose and debug the circuitry of the device using standardized test software. The standardized test software may be reused for every board design or application that uses the embedded processor, and may greatly reduce the time required to 'turn on' a new circuit design.

One of the claimed features is that the standardized test suite is executed by the embedded processor, and not by an outside device. This allows the highest speed testing possible, and allows hardware to be tested separately from the software of a device.

Amendments to the Claims

Claim 1 is amended to include "capable of testing hardware of said single board". Support for the amendment may be found, among other places, in Figure 1 of the specification.

Claim 11 is amended to include "said second software system not able to perform said specific function". Support for the amendment may be found, among other places, in Figure 2 of the specification.

35 USC §102(e) Rejections

Independent Claim 23

Claim 23 is rejected under 35 USC 102(e) as being anticipated by Rohfleisch (US Pat 7,058,855).

The rejection is respectfully traversed for at least the following reasons.

From a recent case before the Court of Appeals for the Federal Circuit (*Net Moneyin, Inc. v. Verisign*, 2007-1565 (Fed. Cir. 2008).):

> Section 102(a) provides that an issued patent is invalid if "the invention [therein] was . . . described in a printed publication . . . before the invention thereof by the applicant." Section 102 embodies the concept of novelty—if a device or process has been previously invented (and disclosed to the public), then it is not new, and therefore the claimed invention is "anticipated" by the prior invention. As we have stated numerous times (language on which VeriSign relies), in order to demonstrate anticipation, the proponent must show "that the four corners of a single, prior art document describe every element of the claimed invention." Xerox, 458 F.3d at 1322 (quoting Advanced Display Sys., Inc. v. Kent State Univ., 212 F.3d 1272, 1282 (Fed. Cir. 2000)). This statement embodies the requirement in section 102 that the anticipating invention be "described in a printed publication," and is, of course, unimpeachable. But it does not tell the whole story. Because the hallmark of anticipation is prior invention, the prior art reference—in order to anticipate under 35 U.S.C. § 102—must not only disclose all elements of the claim within the four corners of the document, but must also disclose those elements "arranged as in the claim." Connell v. Sears, Roebuck & Co., 722 F.2d 1542, 1548 (Fed. Cir. 1983).5
>
> The meaning of the expression "arranged as in the claim" is readily understood in relation to claims drawn to things such as ingredients mixed in some

> claimed order. In such instances, a reference that discloses all of the claimed ingredients, but not in the order claimed, would not anticipate, because the reference would be missing any disclosure of the limitations of the claimed invention "arranged as in the claim." But the "arranged as in the claim" requirement is not limited to such a narrow set of "order of limitations" claims. Rather, our precedent informs that the "arranged as in the claim" requirement applies to all claims and refers to the need for an anticipatory reference to show all of the limitations of the claims arranged or combined in the same way as recited in the claims, not merely in a particular order. The test is thus more accurately understood to mean "arranged or combined in the same way as in the claim."

The Office is required to show every element of the claim "arranged or combined in the same way as in the claim". Applicant asserts that the Office has failed to show:

(1) Claim 23 recites bootstrap code configured to start a software suite on an embedded processor, which is not shown in Rohfleish.

(2) Claim 23 recites a plurality of test workers, which are not shown in Rohfliesh as described in the specification and claims.

(1) Claim 23 recites "bootstrap code". Bootstrap code is code that is executed when a processor starts up, and bootstrap code typically loads other code. In the claims, the bootstrap code is "configured to start a software suite on an embedded processor", and "software suite being capable of executing on said embedded processor".

The software suite as claimed is operated or executed by the embedded processor. This configuration, as described in the specification, allows the hardware of the device to be tested separately from the software. See Figure 5 of the specification and accompanying text in paragraphs [0073] – [0077]. This is a major distinction between the claimed device and Rohfleish.

Rohfliesh describes a system that evaluates the software on the embedded processor. All of Rohfliesh's discussion relates to monitoring a processor, but not replacing the processor's software with test software and running test software on the processor as required in the claim. In

the cited reference of Rohfliesh at Col 4, lines 20-40 and Col 11, lines 10-25, Rohfliesh requires an "emulator circuit".

Rohfliesh at Col 4, lines 10-13 states that "The emulator circuit is operable to receive the respective information indicative of the process activity associated with the at least one processor core and the at least one other core." (Emphasis added).

Rohfliesh does not recognize the problem solved by the claimed invention. The claimed invention separates hardware testing from software testing. Rohfliesh combines software and hardware testing and, in fact, deals mostly with software testing. As described in the background section of the specification at paragraph [0003], "When a new product is being developed, both the hardware and software must be tested and evaluated. In cases where the hardware and software are both complex and the software is required for the hardware to operate, the 'tum on' time for the device may be difficult because problems may not be quickly isolated to either hardware or software."

The claimed invention requires the separation of hardware and software issues by replacing the software on the embedded processor with the test suite and by running the test suite on the embedded processor itself.

Rohfliesh teaches away from separately testing hardware and software. Rohfliesh requires that the embedded processor be running its software so that Rohfliesh's emulator circuit may monitor "process activity associated with the at least one processor core". In the claimed invention, the processor actually runs the software suite, so there is no need for separate "emulator circuits".

(2) Claim 23 recites a plurality of test workers, which are not shown in Rohfliesh as described in the specification and claims.

The claims require "test workers". Among other places, "test workers" are described in paragraph [0006] of the specification: "Specialized test software operating on the embedded processor creates one or more test workers or threads, each having a specific routine to perform. The workers may be executed in parallel, stressing various communication paths."

The Office cites Rohfliesh at Col 12, lines 5-25 as examples of "test workers". Applicant is unsure what features described by Rohfliesh in the cited reference are equivalent to the "test workers" as described in the specification and claims.

The "test worker" as described in the specification and claims is a set of instructions for performing a very specific task.

An example of a "test worker" is given in paragraph [0060] of the specification: "The test workers 306 may be instructions for sending messages across the circuitry to various interfaces. A test worker 306 may contain a destination port, a sequence of data to be sent, any special parameters that can be evaluated, or other information as required."

As claimed, the test workers are executed by the embedded processor. See paragraph [0061] of the specification: "The test workers are run by the processor in block 308. Because the workers are previously created in block 306 and prepared for execution, when the processor runs the test workers in block 308, the messages may be prepared and sent as quickly as possible and thereby stress the circuitry as much as possible."

Applicant requests that the rejection under 35 USC 102 be withdrawn for at least the preceding reasons. Applicant submits that the referenced prior art does not show every element of the claim "arranged or combined in the same way as in the claim" as required for a proper rejection under 35 USC 102.

35 USC §103(a) Rejections

Independent Claim 21

Claim 21 is rejected under 35 USC 103(a) as being unpatentable over DeRolf (US Pat 6,904,544) in view of Narayan et. al. (US 7,010,782).

The rejection is respectfully traversed for at least the following reasons.

(1) The Office has not met its burden under M.P.E.P. § 2143.03 to show cited combination of references that teach a command interpreter adapted to operate on a first embedded processor as claimed in independent claim 21.

(2) The Office has not met its burden under M.P.E.P. § 2143.03 to show cited combination of references that teach a command interpreter adapted to operate on a first embedded processor having a first functionality and a second embedded processor having a second functionality as claimed in independent claim 21.

(3) The Office has not met its burden under M.P.E.P. § 2143.03 to show cited combination of references that teach "said test sequence comprising at least two threads configured to operate in parallel on a single port" as required in independent claim 21.

(4) The Office has not met its burden under M.P.E.P. § 2143.03 to show cited combination of references that teach a command interpreter as claimed in independent claim 21.

(5) The Office has not met its burden under M.P.E.P. § 2143.03 to show cited combination of references that teach "outputting results" as claimed in independent claim 21.

(6) The Office has not met its burden under M.P.E.P. § 2143.03 to show cited combination of references that teach "timestamping on outgoing message" and "timestamping an incoming message" as claimed in independent claim 21.

As stated in the Manual of Patent Examining Procedure at section 2141:

Office Personnel As Factfinders

Office personnel fulfill the critical role of factfinder when resolving the Graham inquiries. It must be remembered that while the ultimate determination of obviousness is a legal conclusion, the underlying Graham inquiries are factual. When making an obviousness rejection, Office personnel must therefore ensure that the written record includes findings of fact concerning the state of the art and the teachings of the references applied. In certain circumstances, it may also be important to include explicit findings as to how a person of ordinary skill would have understood prior art teachings, or what a person of

ordinary skill would have known or could have done. Factual findings made by Office personnel are the necessary underpinnings to establish obviousness.

Once the findings of fact are articulated, Office personnel must provide an explanation to support an obviousness rejection under 35 U.S.C. 103. 35 U.S.C. 132 requires that the applicant be notified of the reasons for the rejection of the claim so that he or she can decide how best to proceed. Clearly setting forth findings of fact and the rationale(s) to support a rejection in an Office action leads to the prompt resolution of issues pertinent to patentability.

In short, the focus when making a determination of obviousness should be on what a person of ordinary skill in the pertinent art would have known at the time of the invention, and on what such a person would have reasonably expected to have been able to do in view of that knowledge. This is so regardless of whether the source of that knowledge and ability was documentary prior art, general knowledge in the art, or common sense.

It is well-established law that, for a proper rejection of a claim under 35 U.S.C. § 103 as being obvious based upon a combination of references, the cited combination of references must disclose, teach, or suggest, either implicitly or explicitly, all elements, features, or steps of the claim at issue. See, e.g., *In Re Dow Chemical*, 5 U.S.P.Q.2d 1529, 1531 (Fed. Cir. 1988), and *In re Keller*, 208 U.S.P.Q.2d 871, 881 (C.C.P.A. 1981). *Glaverbel S.A. v. Northlake Mkt'g & Supp., Inc.*, 45 F.3d 1550, 33 USPQ 2d 1496 (Fed. Cir. 1995) ("the claimed process, including *each step* thereof, *must have been described* or embodied, either *expressly or inherently*.") (Emphasis added.) As clearly articulated in M.P.E.P. § 2143.03, "[to] establish *prima facie* obviousness of a claimed invention, *all the claim limitations must be taught or suggested* by the prior art." *In re Royka*, 490 F.2d 981, 180 U.S.P.Q. 580 (C.C.P.A. 1974). "*All words* in a claim *must be considered in judging the patentability* of that claim against the prior art." *In re Wilson*, 424 F.2d 1382, 1385, 165 USPQ 494, 496 (C.C.P.A. 1970). (Emphasis added.)

The Supreme Court of the United States and the Court of Appeals for the Federal Circuit have provided further guidance for resolving the question of obviousness. "An obviousness

determination is *not the result of a rigid formula* disassociated from the consideration of the facts of a case. Indeed, the common sense of those skilled in the art demonstrates why some combinations would have been obvious where others would not." See *KSR Int'l Co. v. Teleflex Inc.*, 127 S.Ct. 1727, 1739, 167 L.Ed.2d 705 (2007). *Leapfrog Ent., Inc. v. Fisher-Price, Inc.*, 485 F.3d 1157, 82 USPQ2d 1687 (Fed. Cir. 2007). (Emphasis added.)

In *KSR, supra*, the Supreme Court articulated guidelines for determining obviousness. One of the guidelines is that there is some benefit from the proposed modification. ("A person having ordinary skill in the art could have combined Asano with a … sensor …, and *would have seen the benefit* of doing so." *Id*. at 1743 (Emphasis added). "The proper question to have asked was whether … [a person] of ordinary skill [in the art] … *would have seen a benefit to upgrading* Asano with a sensor." *Id*. at 1743 (Emphasis added).)

"Although common sense directs caution as to a patent application claiming as innovation the combination of two known devices according to their established functions, *it can be important to identify a reason that would have prompted a person of ordinary skill in the art to combine the elements as the new invention does.* Inventions usually rely upon building blocks long since uncovered, and claimed discoveries almost necessarily will be combinations of what, in some sense, is already known." *Id*. at 1731 (Emphasis added).

In *KSR, supra*, the Supreme Court noted that it will be necessary for a court "to determine whether there was an apparent reason to combine the known elements in the fashion claimed by the patent at issue. To facilitate review, this analysis should be explicit." *KSR Intern. Co. v. Teleflex Inc.,* 127 S.Ct. 1727, 1741, 1746 (2007). *"[R]ejections* on obviousness grounds *cannot be sustained by mere conclusory statements;* instead, there must be some articulated reasoning with some rational underpinning to support the legal conclusion of obviousness." *Id*. at 1746, quoting *In re Kahn*, 441 F.3d 977, 988 (CA Fed. 2006). (Emphasis added.)

In decisions decided after *KSR, supra*, the Board of Patent Appeals and Interferences has failed to sustain obviousness type rejections when the examiner has failed to make a proper *prima facie* case of obviousness. See *Ex parte Katoh et al*, Appeal 2007-1460 (obviousness rejection not sustained because there was "no evidence or suggestion" in the reference for the alleged configuration); and *Ex parte Crawford et al*, Appeal 2006-2429 (obviousness rejection not sustained because there was no suggestion to combine the references in the manner suggest by

the examiner except for using the Appellants' invention as a template through hindsight reconstruction of Appellants' claims).

(1) The Office has not met its burden under M.P.E.P. § 2143.03 to show cited combination of references that teach a command interpreter adapted to operate on a first embedded processor as claimed in independent claim 21.

As described above, the claimed invention operates on and is executed by an embedded processor. The embedded processor allows the hardware of a device to be fully tested and exercised prior to installing and operating software on the device.

The reusable test sequence as claimed is operated or executed by the embedded processor. This configuration, as described in the specification, allows the hardware of the device to be tested separately from the software. See Figure 5 of the specification and accompanying text in paragraphs [0073] – [0077]. This is a major distinction between the claimed device and DeRolf as well as Narayan.

Both DeRolf and Narayan describe systems that evaluate the performance of a device as a whole, not just the hardware as described in the specification. The claimed invention can only test the hardware because the embedded processor executes the command interpreter and reusable test sequence, not any production code or firmware.

DeRolf describes testing the connections between a Host, through a Fabric, to a Storage Device. (See Fig. 1 of DeRolf). Nowhere does DeRolf describe operating a test sequence on the embedded processor of the device.

Similarly, Narayan describes a system for testing a network device by supplying inputs to the device and monitoring outputs. Nowhere does Narayan describe operating a test sequence on the embedded processor of the network device.

DeRolf or Narayan do not recognize the problem solved by the claimed invention, viewing DeRolf and Narayan separately or in combination. The claimed invention separates hardware testing from software testing. Both DeRolf and Narayan combine software and hardware testing. As described in the background section of the specification at paragraph [0003], "When a new product is being developed, both the hardware and software must be tested and evaluated. In

cases where the hardware and software are both complex and the software is required for the hardware to operate, the 'tum on' time for the device may be difficult because problems may not be quickly isolated to either hardware or software." Also see Figure 5.

The claimed invention requires the separation of hardware and software issues by replacing the software on the embedded processor with the test suite and by running the test suite on the embedded processor itself.

Both DeRolf and Narayan teach away from separately testing hardware and software. Both DeRolf and Narayan require that the embedded processor of the device being tested to be running its software so that the device under test may receive inputs and respond. In the claimed invention, the processor actually runs the software suite, so there is no need for separate host to execute a test sequence.

(2) The Office has not met its burden under M.P.E.P. § 2143.03 to show cited combination of references that teach a command interpreter adapted to operate on a first embedded processor having a first functionality and a second embedded processor having a second functionality as claimed in independent claim 21.

The Office cites DeRolf at Col 3:34-35 "…installed on host systems 2 and 4 to test paths…" for both of the claimed limitations of "a first embedded processor being in a first circuit having a first functionality" and a "second embedded processor being in a second circuit having a second functionality", where "said second functionality being different from said first functionality".

First, in order to show two different functionalities, the Office must actually show two different functionalities. By citing the exact same reference with no explanation, the Office by definition has not shown two different functionalities as claimed.

Second, the claims emphasize the reusable nature of the test sequence by using the same test sequence in two different applications. DeRolf does not show two different applications with two different functionalities, nor does Narayan.

It must be emphasized that, as claimed, the command interpreters operate on different embedded processors and execute the same test sequence. The different embedded processors must be in circuits that have different functionalities.

(3) The Office has not met its burden under M.P.E.P. § 2143.03 to show cited combination of references that teach "said test sequence comprising at least two threads configured to operate in parallel on a single port" as required in independent claim 21.

The Office cites DeRolf at Column 12, lines 20-33 as equivalent. DeRolf at that citation reads:

> In the discussed implementations, the flow of the diagnostic test logic is provided in a rule base object which references descriptors that specify one or more program modules to execute to implement the diagnostic testing. In additional embodiments, different program architectures may be used for the expert diagnostic tool to associate descriptors or program objects with different functions called according to the diagnostic test operations. The diagnostic program may communicate requests for manual operations, e.g., disconnecting, removing and/or replacing components, through a displayable user interface, voice commands, printed requests or any other output technique known in the art for communicating information from a computer system to a person.

As defined in Wikipedia:

> A thread in computer science is short for a thread of execution. Threads are a way for a program to fork (or split) itself into two or more simultaneously (or pseudo-simultaneously) running tasks. Threads and processes differ from one operating system to another but, in general, a thread is contained inside a process and different threads in the same process share some resources while different processes do not. (http://en.wikipedia.org/wiki/Thread_%28computer_science%29 viewed 14 Aug 2008).

Applicant requests that the Office explain how the cited reference is equivalent or even remotely relevant to "said test sequence comprising at least two threads configured to operate in parallel on a single port." Applicant asserts that there is no equivalency between the claimed limitation and the cited reference.

In the specification at paragraph [0006], "Specialized test software operating on the embedded processor creates one or more test workers or threads, each having a specific routine to perform. The workers may be executed in parallel, stressing various communication paths."

While DeRolf may disclose "diagnostic test operations", DeRolf does not describe the architecture and function of the claimed invention. The architecture requires "two threads configured to operate in parallel on a single port", which is not shown in DeRolf.

(4) The Office has not met its burden under M.P.E.P. § 2143.03 to show cited combination of references that teach a command interpreter as claimed in independent claim 21.

In response to the claimed limitation of a "command interpreter", the Office cites DeRolf at Col 3:40-50, "…The rule base code references test descriptors…" as being equivalent to a "command interpreter".

A command interpreter is not described or even mentioned in DeRolf, let alone two instances of a command interpreter as required by the claims.

In the cited reference of Column 3, Lines 40-50, DeRolf uses a 'state machine'. From Wikipedia, "A finite state machine (FSM) or finite state automaton (plural: automata) or simply a state machine, is a model of behavior composed of a finite number of states, transitions between those states, and actions." (http://en.wikipedia.org/wiki/State_machine viewed 14 Aug 2008).

This is in stark comparison to a 'command interpreter', which is defined in Wikipedia as "a computer program that executes, i.e. performs, instructions written in a programming language". Wikipedia further explains that "An interpreter may be a program that either 1. executes the source code directly 2. translates source code into some efficient intermediate representation (code) and immediately executes this 3. explicitly executes stored precompiled code made by a compiler which is part of the interpreter system." (http://en.wikipedia.org/wiki/Interpreter_(computing), viewed 27 Apr 2009).

The 'state machine' of DeRolf is not equivalent to the 'command interpreter' as claimed, and the Office has provided no explanation as to how or why a "state machine" could be interpreted as equivalent to a "command interpreter".

One of the reasons why a 'command interpreter' is claimed is that the test system may be reused in many different applications. By constructing the test system as a command interpreter, the test system may have much more flexibility, configurability, and reusability and a state machine architecture.

(5) The Office has not met its burden under M.P.E.P. § 2143.03 to show cited combination of references that teach "outputting results" as claimed in independent claim 21.

In the cited reference of Column 11, Lines 20-30, DeRolf describes "verbose command causes the state machine to display all messages to a screen display". DeRolf does not state that the 'messages' were results of a test as required in the claim. In fact, DeRolf does not describe displaying any results in any manner.

See Column 11, Lines 40-41, where DeRolf describes an 'interactive' mode:

> interactive: instructs the state machine 102 to allow the user to interact with the state machine to perform manual fault isolation. This arrangement causes the state machine to instruct the administrator to plug and unplug components as the rules evaluate the results to determine the faulty FRU. (Emphasis added).

DeRolf does not describe 'outputting results' as required in the claims. The claims require a specific type of information to be displayed, namely 'results' that were obtained from analysis of timestamps. DeRolf does not describe 'outputting results' of any fashion that may be derived from any test whatsoever.

(6) The Office has not met its burden under M.P.E.P. § 2143.03 to show cited combination of references that teach "timestamping on outgoing message" and "timestamping an incoming message" as claimed in independent claim 21.

The Office cites Column 11, Line 45-55 as being equivalent. DeRolf at that citation reads:

> Once the expert diagnostic tool 100 is invoked with the above arguments, the state machine 102 records a start record with a timestamp into the activity log and processes the rule base completely for each specified disk. When the state machine encounters the end of the rule base, it records the state of the tested

> storage path as COMPLETED or FAILED. If FAILED, the activity log records the name of the loges) that contain failed test data, such as the suspect list 112. These error log files contain important information that should accompany the failed component(s) back to the 55 repair station, such as the suspect list 112 that indicates components that may be the source of the failure. (Emphasis added).

DeRolf only describes timestamping a start time, then processes the rule base completely before timestamping at the end of the rule base. DeRolf as cited does not describe timestamping individual messages that are sent and received as required in the claims.

Applicant respectfully requests the withdrawal of the rejection of claim 21 for at least the preceding reasons.

Independent Claim 22

Claim 22 is rejected under 35 USC 103(a) as being unpatentable over Oberlaender (US PGPUB 2005/0102572) in view of Narayan (US 7,010,782).

The rejection is respectfully traversed for at least the following reasons.

(1) The Office has not met its burden under M.P.E.P. § 2143.03 to show cited combination of references that teach "two threads configured to operate in parallel on a single port". The Office cites "Nagel", but no reference of record is provided that matches the name "Nagel".

(2) The Office has not met its burden under M.P.E.P. § 2143.03 to show cited combination of references that teach "loading said embedded processor with a test platform software comprising…" as required in claim 22. The Office has completely omitted any discussion of this limitation.

(3) The Office has not met its burden under M.P.E.P. § 2143.03 to show cited combination of references that teach assembling a circuit as claimed in independent claim 22.

(4) Oberlaender is non-analogous art and does not teach of operating a processor as claimed. Oberlaender only teaches of simulating a processor, not executing an actual processor in

an actual circuit as claimed. Further, the claimed invention is directed towards testing hardware, not software.

(5) The Office has not met its burden under M.P.E.P. § 2143.03 to show cited combination of references that teach "a command interface" as claimed in independent claim 22.

(6) The Office has not met its burden under M.P.E.P. § 2143.03 to show cited combination of references that teach "timestamping outgoing messages and storing said messages" as claimed in independent claim 22.

(7) The Office has not met its burden under M.P.E.P. § 2143.03 to show cited combination of references that teach "displaying the results" as claimed in independent claim 22.

(8) The Office has not met its burden under M.P.E.P. § 2143.03 to show cited combination of references that teach "transmitting a test sequence to an embedded processor" as claimed in independent claim 22.

(1) The Office has not met its burden under M.P.E.P. § 2143.03 to show cited combination of references that teach "two threads configured to operate in parallel on a single port". The Office cites "Nagel", but no reference of record is provided that matches the name "Nagel".

Narayan at the cited Col. 11, lines 43-50 state: "For example, ten ports of switch 62 connect to ten ports of switch 64, and another ten ports of switch 62 connect to ten ports of switch 70. The network topography of the test bed is to check for convergence of the switches using a specific protocol which ensures and produces result that the network converges at a point where N number of virtual local-area-network (vlan) configurations is added to the network. The network convergence is measured by the output that is being generated when the traffic generator produces traffic."

Applicant is unable to find any notion of "threads", let alone "threads configured to operate in parallel on a single port" in this reference.

Neither Oberlaender, DeRolf, nor any reference of record mentions having two threads operating in parallel on a single port. The operation of two or more threads on a single port may enable the full bandwidth of the port to be used up, which may enable various performance measurements.

(2) The Office has not met its burden under M.P.E.P. § 2143.03 to show cited combination of references that teach "loading said embedded processor with a test platform software comprising…" as required in claim 22.

The Office has completely omitted any discussion of this limitation. Applicant is unsure how to respond, and respectfully requests that if the Office continues in the rejection of this claim that such a rejection be a Non-Final Office Action, as a Final Office Action is precluded under MPEP 706.07(a).

The limitation of "loading said embedded processor with a test platform software" emphasizes the architecture and function of the claimed invention. The claimed invention operates on and is executed by an embedded processor of a device under test. By running the "test platform software" on the embedded processor, the hardware of a device can be fully tested and exercised prior to installing and operating software on the device, where the software enables the device to perform its intended function.

In claim 22, the embedded processor must also have "software adapted to enable said circuit to perform said predefined function", which is separate and distinct from the "test platform software" that is loaded and executed on the embedded processor. These two limitations emphasize that the circuit has an intended and "predefined function" that is implemented by the "software", but the "test platform software" can be operated on the embedded processor separately.

The reusable test sequence as claimed is operated or executed by the embedded processor. This configuration, as described in the specification, allows the hardware of the device to be tested separately from the software. See Figure 5 of the specification and accompanying text in paragraphs [0073] – [0077]. This is a major distinction between the claimed device and all of the cited references, including Narayan, and Oberlaender.

Both Oberlaender and Narayan describe systems that evaluate the performance of a device as a whole, not just the hardware as described in the specification. The claimed invention can only test the hardware because the embedded processor executes the command interpreter and reusable test sequence, not any production code or firmware.

Oberlaender describes testing memory contents of a System On a Chip by using a circuit simulation system. Oberlaender solves a problem of monitoring vast amounts of memory changes when software operates on the System On a Chip. The entire focus of Obelaender is to evaluate the software operating on the System On a Chip, not the hardware.

Narayan describes a system for testing a network device by supplying inputs to the device and monitoring outputs. Nowhere does Narayan describe operating a test sequence on the embedded processor of the network device.

Oberlaender and Narayan do not recognize the problem solved by the claimed invention, viewing Oberlaender and Narayan separately or in combination. The claimed invention separates hardware testing from software testing. Both Oberlaender and Narayan combine software and hardware testing. As described in the background section of the specification at paragraph [0003], "When a new product is being developed, both the hardware and software must be tested and evaluated. In cases where the hardware and software are both complex and the software is required for the hardware to operate, the 'tum on' time for the device may be difficult because problems may not be quickly isolated to either hardware or software." Also see Figure 5.

The claimed invention requires the separation of hardware and software issues by replacing the software on the embedded processor with the test suite and by running the test suite on the embedded processor itself.

Both Oberlaender and Narayan teach away from separately testing hardware and software. Both Oberlaender and Narayan require that the embedded processor of the device being tested to be running its software so that the device under test may receive inputs and respond. In the claimed invention, the processor actually runs the software suite, so there is no need for separate host to execute a test sequence.

(3) The Office has not met its burden under M.P.E.P. § 2143.03 to show cited combination of references that teach assembling a circuit as claimed in independent claim 22.

Oberlaender does not actually teach of assembling a circuit, but only of fabricating a single integrated circuit device in paragraph [0003]. In order to 'assemble', one must take two or more distinct pieces and join them together. This differs from the manufacture of integrated circuits where various layers are built up as defined in masks.

The distinction between 'assembling' a circuit as claimed and the fabrication of a circuit of Oberlaender highlights some of the differences between the claimed invention and the cited prior art. The claimed invention describes method for developing a circuit, such as a circuit on a printed circuit board, where an embedded processor may be loaded with a test sequence that fully exercises the circuitry.

The Office further cites paragraph [0025] and states "it is here that Oberlaender teaches that the simulation model includes a processor core that can process program instructions that are associated with test programs and data file that are stored in the memory array. Later on Oberlaender mentions that separate program and data memory devices may be used. Therefore, Oberlaender does teach software operable on said embedded processor."

Applicant respectfully asserts that this citation teaches away from the claimed invention as interpreted by the Examiner. The claimed invention requires that the test software operate on the embedded processor, not an external or host processor.

(4) Oberlaender is non-analogous art and does not teach of operating a processor as claimed. Oberlaender only teaches of simulating a processor, not executing an actual processor in an actual circuit as claimed. Further, the claimed invention is directed towards testing hardware, not software.

Oberlaender uses simulated circuits and processors, not actual hardware. Unless the simulation is incorrect or inaccurate, simulated hardware always functions properly and never needs to be tested. Because the claimed invention can only test hardware, and Oberlaender can only test software, using the claimed invention in the context of Oberlaender would be a pointless exercise.

The claimed invention is directed at testing hardware only, not the software that operates on the hardware. The operable software that would have been tested by Oberlaender is removed from the device and replaced with the claimed test platform software.

(5) The Office has not met its burden under M.P.E.P. § 2143.03 to show cited combination of references that teach "a command interface" as claimed in independent claim 22.

As claimed, the 'command interface' is 'adapted to receiving commands and outputting results'. The Office cites Oberlaender's 'interface circuit' as described in paragraph [0027] as equivalent.

Oberlaender's 'interface circuit' is item 222. From paragraph [0025] of Oberlaender: "In addition, memory array 220 includes an interface circuit 222 that communicates with CPU pipeline 210 over system bus 215, an array of memory cells 226, and a memory control unit (MCU) 224 coupled between interface circuit 222 and memory cells 226." Oberlaender's 'interface circuit' is not 'adapted to receiving commands and outputting results' as claimed, but only serves to connect different components within a simulated circuit.

(6) The Office has not met its burden under M.P.E.P. § 2143.03 to show cited combination of references that teach "timestamping outgoing messages and storing said messages" as claimed in independent claim 22.

Outgoing and incoming messages as described in the specification, are messages sent through input and output ports on a device with an embedded processor.

The Office cites Oberlaender's 'timestamp values' as equivalent. Oberlaender discusses timestamping only in the context of recognizing a change in a simulated memory device and storing the memory information with a time stamp.

The cited reference includes: "To chronical (sic) simulated accesses to memory array 220 (e.g., simulated data write operations), address and data signals transferred to interface circuit 222 are captured by memory tracer 234, which then generates incremental transaction records that are stored in transaction record storage region 236 (block 340). As described in additional detail below, each transaction record includes a timestamp value indicating when the transaction occured, address information identifying, for example, the memory cells changed during a write operation, the actual data written to these memory cells, and, an optional data field indicating an access width (i.e., amount of the accessed/written data)." The "transaction record" is not equivalent to a "message" as claimed.

Oberlaender does not teach of timestamping incoming or outgoing messages as required in the claims.

(7) The Office has not met its burden under M.P.E.P. § 2143.03 to show cited combination of references that teach "displaying the results" as claimed in independent claim 22.

The Office cites Oberlaender at paragraph [0063] and specifically the sentence "Data is typically written to fifo devices using "push" or "push write" operations, and is read removed) from the fifo device using "pop" operations." The Office provides no further explanation as to why this sentence has anything to do with "displaying the results" as claimed.

Absent any explanation by the Office, Applicant asserts that "displaying the results" of a performance test has nothing to do with the operation of FIFO devices.

(8) The Office has not met its burden under M.P.E.P. § 2143.03 to show cited combination of references that teach "transmitting a test sequence to an embedded processor" as claimed in independent claim 22.

The Office cites Oberlaender at paragraph [0048] as being equivalent, specifically a "…parallel shows the transaction stream…". In paragraph [0048], Oberlaender discusses a GUI 238 (Graphical User Interface) that "displays the instantaneous data values stored in the memory array, and in parallel shows the transaction stream leading up to the instantaneous memory content (i.e., one or more transaction records associated with the displayed instantaneous data values)." Oberlaender goes on to state that an exemplary representation is shown in Figure 7.

In paragraph [0048] and Figure 7, Oberlaender does not teach nor even mention 'transmitting a test sequence to an embedded processor'. Oberlaender only discusses a user interface that may show transaction records and values in a memory array.

The claims require that a test sequence be 'transmitted to an embedded processor'. The Office has failed to cite any transmission and further failed to cite any embedded processor as required in claim 22.

Applicant respectfully requests the withdrawal of the rejection of claim 22 for at least the preceding reasons.

Independent Claim 1

Claims 1-7, 9 and 10 are rejected under 35 USC 103(a) as being anticipated by DeRolf in view of Toth (US Pat 4,829,520).

The rejection is respectfully traversed for at least the following reasons.

(1) The Office has not met its burden under M.P.E.P. § 2143.03 to show cited combination of references that teach "timestamping on outgoing message" and "timestamping an incoming message" as claimed in independent claim 1.

The Office cites Column 11, Line 45-55 as being equivalent. DeRolf at that citation reads:

> Once the expert diagnostic tool 100 is invoked with the above arguments, the state machine 102 records a start record with a timestamp into the activity log and processes the rule base completely for each specified disk. When the state machine encounters the end of the rule base, it records the state of the tested storage path as COMPLETED or FAILED. If FAILED, the activity log records the name of the loges) that contain failed test data, such as the suspect list 112. These error log files contain important information that should accompany the failed component(s) back to the 55 repair station, such as the suspect list 112 that indicates components that may be the source of the failure. (Emphasis added).

DeRolf only describes timestamping a start time, then processes the rule base completely before timestamping at the end of the rule base. DeRolf as cited does not describe timestamping individual messages that are sent and received as required in the claims.

(2) The Office has not met its burden under M.P.E.P. § 2143.03 to show why the cited references of DeRolf and Toth are combined.

DeRolf does not recognize the problem solved by the claimed invention. The claimed invention separates hardware testing from software testing. DeRolf combines software and hardware testing. As described in the background section of the specification at paragraph [0003],

"When a new product is being developed, both the hardware and software must be tested and evaluated. In cases where the hardware and software are both complex and the software is required for the hardware to operate, the 'tum on' time for the device may be difficult because problems may not be quickly isolated to either hardware or software." Also see Figure 5.

Toth does not recognize the problem solved by the claimed invention. Toth has a routine in an embedded processor that stores diagnostic data for a single board device so that the board may be diagnosed when serviced. The software used by Toth is the operational software for the device. The claimed invention allows hardware testing from a command interface by timestamping incoming and outgoing messages. Toth does not discuss timestamping or a command interface, for example.

Neither DeRolf nor Toth give any reason why the remote testing mechanism of DeRolf should be combined with the onboard testing of Toth, because remote testing is generally mutually exclusive of or at least separate from onboard testing.

Applicant respectfully requests the withdrawal of the rejection of claim 1 for at least the preceding reasons.

Independent Claim 11

Claim 11 is rejected under 35 USC 103(a) as being anticipated byToth in view of DeRolf.

The rejection is respectfully traversed for at least the following reasons.

(1) The cited combination of references do not teach "a first software system…having a specific function" and "a second software system not able to perform said specific function…".

Toth describes a printed circuit board with an embedded processor, but with an operating program that performs some diagnostics. However, neither Toth nor DeRolf describe the second software system that replaces the first software system, where the second software system may perform tests as described in the claims.

Applicant respectfully requests the withdrawal of the rejection of claim 11 for at least the preceding reasons.

Dependent Claims 7, 16, 17, and 19

Claims 7, 16, 17, and 19 are rejected under 35 USC 103(a) as being anticipated by DeRolf in view of Toth.

(1) The Office has not met its burden under M.P.E.P. § 2143.03 to show cited combination of references that teach "at least two of said multiple threads of commands being configured to operate on a single port" as required in dependent claims 7 and 16

The Office cites Column 13, lines 15-20 of DeRolf as being equivalent. The cited reference reads: "determining a path in the system to test, wherein the path includes path components including at least a host adaptor, a link, a device interface, and a device, wherein the device comprises a storage system; performing an initial test to determine if there is a failure in the path."

As defined in Wikipedia:

> A thread in computer science is short for a thread of execution. Threads are a way for a program to fork (or split) itself into two or more simultaneously (or pseudo-simultaneously) running tasks. Threads and processes differ from one operating system to another but, in general, a thread is contained inside a process and different threads in the same process share some resources while different processes do not. (http://en.wikipedia.org/wiki/Thread_%28computer_science%29 viewed 14 Aug 2008).

DeRolf does not mention threads in any place whatsoever, and Toth does not overcome the deficiency of DeRolf.

As explained above, DeRolf uses a state machine which is not a multithreaded interpreter, which would be required as claimed. In addition to the fact that DeRolf does not in any manner describe multiple threads, it would be technically impossible to have a test sequence comprising multiple threads of commands using a state machine as taught by DeRolf.

The claims require threads and specifically multiple threads. DeRolf does not specifically mention threads, thread based architectures, or any other technology that may be construed as containing threads. While DeRolf does attempt an omnibus statement in column 12, lines 20-33, all of DeRolf's various embodiments relate to different descriptors for functions (column 12, lines 20-26), different methods for user interface interaction (column 12, lines 27-32), and different architectures for Storage Area Networks (column 12, lines 33-40). DeRolf does not, in any manner, describe different architectures for the test mechanism other than the state machine.

(2) The Office has not met its burden under M.P.E.P. § 2143.03 to show cited combination of references that teach "an input driver further adapted to validate an incoming message" as claimed in dependent claim 17.

The Office cites DeRolf at column 13, lines 15-20 "a device interface". The citation states a 'device interface', but does not attribute any specific functionality to the 'device interface'. Specifically, the claim requires that the "an input driver further adapted to validate an incoming message". The Office has not shown where the 'device interface' validates an incoming message, nor has the Office explained where such a feature may be attributed.

(3) The Office has not met its burden under M.P.E.P. § 2143.03 to show cited combination of references that teach "an initiator adapted to determine if an I/O device is present" as claimed in dependent claim 19.

The Office cites DeRolf at column 13, lines 15-20 "a device interface". The citation states a 'device interface', but does not attribute any specific functionality to the 'device interface'. Specifically, the claim requires that the "an initiator adapted to determine if an I/O device is present". The Office has not shown where the 'device interface' validates an incoming message, nor has the Office explained where such a feature may be attributed.

Applicant respectfully requests the withdrawal of the rejection of claims 7, 16, 17, and 19 for at least the preceding reasons.

In view of the foregoing, Applicant respectfully submits that the independent claims patentably define the claims over the citations of record. Further, the dependent claims should also be allowable for the same reasons as their base claims and further due to the additional features that they recite and for the separate arguments presented for the dependent claims. Separate and individual consideration of the dependent claims is respectfully requested.

Applicant believes no new material has been added. Applicant invites the Examiner to call the undersigned at 970-690-4023 to suggest any changes or discuss any actions so that this case may be quickly allowed.

The applicant believes the application to be in condition for allowance, and such action is earnestly requested.

Dated this _27th_ day of __April__, 2009.

Respectfully submitted:

KRAJEC PATENT OFFICES, LLC

__/Russell S. Krajec/_____
Russell S. Krajec
Reg. No 48,936
Krajec Patent Offices, LLC
820 Welch Ave
Berthoud, Colorado 80513
970.690.4023
970.690.4074 (f)